



RAPPORT RV01 - A12

Projet présenté au 3DUI Contest 2013





Marie COLLET
Esteban PASQUIER
Loïc PICAVET
Arthur VAN CEULEN





Sommaire

Contexte

Dans le cadre de l'UV RV01, Réalité Virtuelle, nous avons l'objectif d'appliquer les techniques et théories apprises en cours dans un projet tout au long du semestre. Dans ce contexte, il nous a été proposé de participer au concours 3DUI 2013. Celui-ci consiste à la création d'un environnement virtuel dans lequel l'utilisateur sera capable de créer, à partir de rien, son propre monde et d'interagir avec ce dernier. Le concours imposant que le monde soit dans l'univers de Disney, nous avons choisi le thème d'Alice aux Pays des Merveilles, permettant une grande liberté dans la création du monde.

Notre équipe est constituée de 10 personnes :

Etudiants Ingénieurs

PICAVET Loïc
PASQUIER Esteban
COLLET Marie
VAN CEULEN Arthur

Doctorants

CARPENTIER Kévin
WAGREZ Kévin
BAROT Camille
LANQUEPIN Vincent

Etudiants Master UX Design

MULLER Mathilde
CUELLA-MARTIN Andrea

Enseignante UTC
MOUTTAPA-THOUVENIN Indira

Objectifs

L'utilisateur devra être capable de créer son propre monde et dans un temps limité de quinze minutes, défini par le concours. Pour cela, la création des objets devra être simple et intuitive, et les interactions de l'utilisateur avec ces derniers et le monde ne devront nécessiter qu'un minimum d'apprentissage. Nous nous sommes pour ce faire inspirés de la vidéo de présentation du concours 3DUI : dans celle-ci, un menu parfaitement intégré dans l'univers permet la création et la modification (couleurs, textures) des objets, et ce en toute simplicité grâce à un menu intuitif et ergonomique. Nous souhaitons mettre en place ce type de menu pour notre monde, en y intégrant une modalité d'interactions différente mais innovante. En ce qui concerne l'interaction avec les objets et également le monde, l'objectif est d'utiliser des gestes naturels à la fois pour se déplacer dans le monde et pour modifier les objets créés, tout comme il est possible de le voir dans la vidéo : écartement des mains pour agrandir, rotation pour tourner l'objet, application de couleur ou textures par simple toucher, etc.

Réalisation

Interaction vocale

Au moment de la conception, nous avons réfléchi à une solution innovante qui nous permettrait d'interagir avec l'application de manière simple tout en gardant les mains libres pour manipuler le monde.

C'est assez naturellement que nous nous sommes tournés vers la navigation par commande vocale qui devient de plus en plus présente et performante (SIRI sur Iphone, Google voice sur android, etc…). Cela n'avait jamais été fait dans le cadre du cours RV01 et le contest 3DUI demandait une dimension innovante, la reconnaissance vocale semblait être le choix idéal.

Solution

Après un état de l'art, nous nous avons retenu deux technologies : une API Java "Java Text To Speech" et la reconnaissance vocale de windows 7. Cette dernière a retenu notre attention par la pertinence des résultats obtenus : nous avons été bluffés car nous n'avons obtenu que très rarement des erreurs de compréhension.

Étape 1 : Récupération des mots

Nous avons donc développé une application permettant de récupérer les mots énoncés par l'utilisateur grâce à l'API windows (Windows 7 Speech Recognition Engine).

Dans cette application nous chargeons deux fichiers (les objets du monde + les interactions avec le monde) qui constituent ainsi la grammaire, c'est à dire l'ensemble des mots que peut reconnaître l'application.

Cette application a été développée avec la technologie C#.

Etape 2 : Transmission des mots vers Virtools

Une fois les mots captés nous devons maintenant les transmettre à l'application Virtools. Le logiciel permet l'utilisation du C++ afin de créer nos propres Building blocks. C'est ce que nous avons fait. Nous en avons développé un permettant de recevoir, sous forme d'un String, les mots prononcés et captés dans l'application précédente.

Une fois le mot reçu dans Virtools, il suffit de le diffuser en utilisant le building block "Broadcast Message".

Utilisation

Le menu se présente sous forme de bulles flottantes dans l'environnement. Elles sont placées toujours face à la vision de l'utilisateur. Les mots à prononcer pour effectuer l'action sont affichés dans la bulle. A la reconnaissance du mot, la bulle éclate, pour un feedback cohérent pour l'utilisateur.

Menu

Le menu général permet la création d'objets répartie sous forme de hiérarchie (exemple: Environment -> Builds-> Cube). Mais aussi la mise en place de différents ciels via le menu skybox.

L'organisation hiérarchique donne ainsi la possibilité de descendre ou monter d'un niveau à la prononciation du mot de la bulle (le nœud) respectivement à la reconnaissance du mot "back".

Le dernier niveau du menu correspond aux objets à créer. Pour ce niveau, nous avons ajouté en plus des mots à prononcer des aperçus de l'objet dans la bulle, permettant de rendre compte de l'aspect de l'objet et d'augmenter l'intuitivité du menu.

Menu Contextuel

Il permet la manipulation des objets (move, rotate, scale) et leur texturing (Textures et Couleurs). Ce menu apparaît lorsqu'un objet est sélectionné, celui-ci l'est en pointant un objet grâce au Wand, que nous verrons dans la partie qui suit, puis en prononçant le mot "this". Nous avons également ajouté un retour visuel sur cette sélection, en effet l'objet sera entouré de particules blanches. Pour revenir sur le menu contextuel, celui-ci affiche, toujours sous forme de bulles, les possibilités d'action sur l'objet sélectionné. Nous avons donc 3 bulles pour la manipulation, 2 pour l'habillage de l'objet et enfin 2 autres, "exit" et "stop", respectivement pour dé-sélectionner l'objet et stopper l'action en cours. Pour d'avantage d'informations sur ce menu, un manuel d'utilisateur est disponible.

Interaction Spatiale

L'utilisateur est muni du HeadMan Display de Sony qui lui permet d'être immergé dans le monde virtuel 3D.

Virtools inclut le module VRPN, permettant de gérer les périphériques de réalité virtuelle, et nous avons donc intégré les divers trackers (constellations qui seront accrochées sur les parties du corps de l'utilisateur) que nous avons calibrés à travers des fichiers de configuration. Le VRNR permet quant à lui d'offrir des interactions classiques de la réalité virtuelle, et nous notamment l'avons utilisé pour la vision subjective et la visée au wand.

Déplacement de l'utilisateur

L'utilisateur se déplace à l'aide de deux constellations placées sur chacun de ces deux pieds. Il avancera le pied droit pour avancer, et reculera ce même pied pour reculer.

Lorsque les deux pieds sont à une distance de 35cm ou plus, le déplacement s'effectue. Pour aller plus vite, l'utilisateur pourrait choisir d'avancer plus le pied, mais cela étant en réalité moins intuitif que l'on pourrait le penser, c'est grâce à la reconnaissance vocale que se fera l'accélération : quand l'utilisateur prononcera le mot "faster", ou "plus vite" dans la version française, ce qu'il pourra répéter indéfiniment pour être de plus en plus rapide.

Orientation de l'utilisateur

Nous avons choisi de désactiver le déplacement automatique déjà scripté dans le VRNR, qui permettait un déplacement naturel de l'utilisateur dans la pièce. Mais notre utilisateur est muni d'un casque Sony et ne voit donc pas les obstacles présents dans la pièce, et de plus la taille du monde se retrouvait très restreinte avec ce choix. Nous avons donc imaginé que l'utilisateur pourrait se déplacer en avançant un pied par rapport à l'autre, et s'arrêter en joignant les deux pieds.

Le choix de l'orientation de l'utilisateur dans l'espace lors de son déplacement a été possible après réception des résultats des tests utilisateurs. Le premier choix était de permettre à l'utilisateur un déplacement selon le vecteur entre ses deux pieds, ce qui lui permettrait donc d'avancer tout droit tout en tournant la tête à droite ou à gauche. Ce mode de déplacement semble plus immersif car plus proche de celui de l'homme dans la réalité, mais les tests utilisateurs ont révélé qu'il n'était pas très intuitif, ni très facile à l'utilisation experte. Le choix s'est donc porté sur un déplacement selon l'axe du champ de vision, porté par le VRNR_Navigation_NavHead grace à une constellation placée au dessus du casque Sony.

Changement de taille de l'utilisateur

L'utilisateur pourra choisir de se grandir ou de se rétrécir par rapport à la taille du monde, en prononçant "make me big!" ou "make me small!" comme le fait Alice en mangeant un gâteau. Cette fonction lui permet alors de créer des objets plus grands ou plus petits en fonction de sa taille dans le monde, et également de se déplacer plus vite ou plus lentement. Il ne peut toutefois pas s'agrandir et se rétrécir à l'infini : il a trois tailles possibles, une taille par défaut (à l'échelle du monde), une taille trois fois plus grande et une taille trois fois plus petite. C'est à l'aide d'une translation du Nav_Center ainsi qu'une variable globale "size" que nous avons développé cette fonction.





Manipulation des objets

Sélection

Pour sélectionner un objet, l'utilisateur le vise avec le rayon du wand et prononce le mot "this". L'objet sera alors sélectionné comme le montre le retour visuel de la surbrillance blanche, gérée à l'aide de particules par le BB Object Particle System. Le menu contextuel s'ouvrira. Pour quitter la sélection, il suffit de prononcer "exit" ou de sélectionner un autre objet avec "this".

Mouvement

Pour déplacer un objet, celui-ci doit préalablement être sélectionné et une bulle suggère alors le mot "move". Lorsqu'il est détecté, ce mot déclenche le début du mouvement qui peut être arrêté par "stop", ou bien "exit" qui dé-selectionnera directement l'objet. L'utilisateur peut déplacer l'objet autour de lui avec le wand, puisque celui-ci est dans le référentiel du wand : l'objet a donc toujours la même face par rapport à l'utilisateur, ainsi qu'une distance constante. Mais le déplacement ne se comporte pas exactement ainsi puisque nous n'autorisons que les rotations du plan : lorsque l'utilisateur cherche à monter un objet (déplacement suivant l'axe y), celui-ci ne tournera pas vers lui mais restera droit. Cela est bien plus convenant pour le déplacement des objets, c'est une idée que nous avons tirée du célèbre jeu vidéo Portal.

Création

Après avoir navigué dans le menu et avoir prononcé le mot de l'objet à créer, l'utilisateur voit l'objet apparaître à une distance constante de lui-même, sur le rayon du wand, comme nous le conseillaient les retours des tests utilisateur : "créer un objet là où on vise". L'objet est déjà sélectionné après la création, et est déjà en phase de mouvement. Ainsi, il peut directement le déplacer à l'endroit où il le veut.

Couleurs / textures

L'utilisateur peut prononcer le mot "color" ou "texture" depuis le menu contextuel d'un objet sélectionné pour naviguer dans un sous-menu qui va proposer différentes couleurs ou textures. En prononçant le mot souhaité, on applique une couleur, ou une texture qui sera mappée selon la bounding box et la taille de l'objet. Il est possible de mélanger une couleur avec une texture et nous avons donc utilisé des textures en niveaux de gris pour avoir un rendu correct en invitant à utiliser cette fonctionnalité.

Rotation

Après sélection d'un objet, une bulle suggère le mot rotate qui une fois prononcé ne fournit qu'un retour sonore, mais la position du wand à cet instant est enregistrée. Seule la rotation autour de l'axe y est possible, et elle s'active dans un sens ou dans l'autre lorsque l'on déplace le wand vers sa gauche ou vers sa droite. En fait, passée une certaine distance, la rotation s'active à vitesse constante, et lorsque l'on repasse le seuil de distance, elle s'arrête.

Redimensionnement

Le scale fonctionne de la même manière que le rotate pour son activation, mais la différence est que l'on peut l'activer suivant les 3 axes, selon les 2 sens, et selon plusieurs axes à la fois si nécessaire. Les mouvements selon l'axe x déforme l'objet parallèlement à l'utilisateur, et ceux selon l'axe z perpendiculairement à l'utilisateur. A la fin d'un scale, on map la texture à nouveau de manière à ne pas avoir un rendu trop laid (en particulier pour éviter l'étirement de la texture).

Le monde virtuel

Certains objets ont été réalisés sous 3DS Max ; d'autres libres de droits ont été récupérés sur internet, tout comme les textures. Les modèles ont été choisis pour coller au maximum à l'univers d'Alice au Pays des Merveilles. L'utilisateur peut changer les couleurs des objets : il dispose pour cela d'une palette de cinq couleurs correspondant à celles du dessin animé.

L'utilisateur peut également changer le ciel grâce à des skybox réalisées avec le logiciel Terragen. Les six faces du ciel sont chargées quand l'utilisateur choisit celui qu'il désire dans le menu. Il a le choix entre le jour, la nuit, un coucher de soleil et une nuit étrange.

A son entrée dans le monde, l'utilisateur est sur un chemin cerné par des murs, ce qui va lui permettre de se familiariser avec le déplacement avant d'entrer dans une clairière délimitée par des montagnes et des murailles. Il pourra dès lors créer et manipuler des objets.





Bilan général

Difficultés rencontrées

La gestion de projet n'a pas été simple, cela compte tenu du grand nombre de membres composant l'équipe. Elle était composée de dix personnes et notamment de quatre doctorants dont il fallait naturellement respecter les décisions.

Nous avons fait l'erreur de ne pas utiliser de gestionnaire de versions, ce qui nous convenait bien au début du projet quand les tâches étaient bien réparties entre deux binômes de RV01, mais le projet s'est complexifié lors de la fusion du menu et des autres interactions. Dès lors, nous étions potentiellement huit développeurs à travailler sur la même version de projet, et ceci dans un temps très limité. Un gestionnaire de tâches aurait aussi nécessairement utile, et nous avons d'ailleurs parfois réalisé deux fois le même travail à cause d'incompréhensions ou de manque de communication. La répartition des tâches était d'ailleurs plutôt claire au début du semestre, mais plus la date de l'échéance s'est approchée et plus les tâches se sont multipliées, la répartition s'est alors faire de manière parfois logique, parfois aléatoire.

La taille du groupe et la diversité des personnes le constituant (GI01 à doctorant) avec des emplois du temps très différents nous ont parfois fait perdre un temps conséquent dans la prise de décision : lorsqu'un choix se posait, il fallait attendre la réunion souvent hebdomadaire pour en discuter, et parfois la réunion elle-même était insuffisante car il fallait attendre des résultats préliminaires de tests utilisateur, les résultats étant ensuite encore à discuter en réunion. L'autorité des doctorants dans la prise des décisions n'était pas non plus claire et nous ne savions pas vraiment si nous pouvions imposer certains choix qui étaient normalement sous notre responsabilité. Par exemple, il fallu près de 3 semaines pour décider du déplacement selon l'axe de vision de l'utilisateur, au lieu que ce soit selon l'axe formé par les deux pieds. Ce sont les tests utilisateurs qui ont permis ce choix.

La taille de l'application nous a amené à la développer par "morceaux", afin de pouvoir travailler plusieurs aspects parallèlement. Cela a créé de nombreux problèmes lors des merges : parfois les scripts ne fonctionnaient plus du tout car par exemple les items créés étaient en fait des characters, ou étaient composés de plusieurs sous-objets ou de plusieurs meshs ; c'était pourtant pour régler un autre problème que l'on avait fait le choix de mettre tous les objets ainsi !

En outre, le snapping (magnétisme qui colle les objets sur des surfaces adjacentes afin d'aider leur placement) qui a été développé pour des objets cubiques est très fonctionnel, mais il a été quasiment impossible de l'intégrer au système complet : trop de cas possibles apparaissent tellement les objets sont modifiables, alors que le script de base est déjà extrêmement complexe.

Cette généricité, contrainte du concours, a d'ailleurs été assez problématique également : nous avons toujours cherché à être exhaustif dans les possibilités offertes (catalogue d'objets important, toutes les couleurs, que les objets n'aient jamais de "bas" défini car tout objet peut être retourné, etc…) et cela nous a coûté beaucoup de temps. En plus, vu le temps restant et pour l'aspect "scenario driven", nous n'avons finalement pas respecté certaines formes de généricité comme par exemple les couleurs ou le "rotate" selon tous les axes.





Le concours imposait l'utilisation de la langue anglaise, en cela, nous avons eu quelques difficultés autour de la reconnaissance vocale. En effet, la prononciation non parfaite empêchait parfois la reconnaissance de certains mots. En fin de projet, nous avons, sans difficultés, ajouté la possibilité de naviguer dans le menu à l'aide de la langue française, l'interaction avec le monde en était que facilité.

Un second problème est survenu concernant la reconnaissance vocale, en effet, possédant des versions française de Windows 7, il nous a fallu installer un module de reconnaissance pour la langue anglaise. Il se trouve que pour des raisons inconnues, ce module pouvait se désinstaller du système, certainement une limitation de Microsoft. Ce problème nous obligeait à ré-installer systématiquement le module ; cela ne prenait que quelque minutes, mais de précieuses minutes en salle immersive, et fallait-il encore savoir que cette manipulation était nécessaire, lorsque le responsable de la commande vocale n'était pas présent.

Ce problème et bien d'autres insignifiants, ajoutés à la complexité de notre système, ont conduit à une perte de temps systématique à chaque relance du système.

Résultats par rapport à l'objectif

Nos objectifs semblent avoir été remplis. En effet, l'utilisateur est capable de créer des objets, de modifier leur taille, leur orientation, leur position, mais aussi leur texture et leur couleur, et tout cela grâce à des interactions à la fois innovantes et intuitives : la reconnaissance vocale combinée au tracking de la personne permettent à l'utilisateur de construire son monde virtuel sans difficultés. Ces interactions naturelles diminuant la phase d'apprentissage du système et renforçant l'immersion dans le monde, permettent également de créer rapidement son monde et de respecter ainsi le temps limité pour la construction.

Retours utilisateurs et améliorations possibles

Les utilisateurs ont dans l'ensemble trouvé notre système sympathique, amusant et intuitif. Les interactions développées sont pertinentes et d'une fluidité acceptable et l'interface est ergonomique: la navigation dans le menu est intuitive et permet à l'utilisateur de trouver rapidement l'outil qu'il cherche car il n'a que deux outils à disposition: sa voix et sa baguette.

Certains détails sont cependant en attente d'amélioration. En effet nous avons été pris de court par le délai assez serré, ce qui nous a obligés à faire des concessions.

Nous n'avons pas pu intégrer le snapping des objets. Cette fonction a été développée en partie, par exemple pour coller deux cubes. Nous aurions aimé aller plus loin en faisant du cas par cas et laisser à l'utilisateur la possibilité de coller des objets de manière très intuitive, mais cette étape nous prenait beaucoup de temps et nous avons préféré nous concentrer sur d'autres priorités.

Nous aurions aimé différencier clairement un mode normal d'un mode expert, qui permettrait à l'utilisateur de créer des objets sans passer par tout le menu, donc en prononçant juste le nom de l'objet. Dans le cas du mode normal, la reconnaissance des mots par l'API se ferait uniquement sur les mots présents dans les bulles à ce moment donné, enlevant ainsi toute ambiguïté possible entre deux mots du système.





Conclusion générale

La diversité des membres composant cette équipe nous a permis de travailler sérieusement la plupart des aspects, et de réaliser une application répondant à nos critères, cela en un temps très limité. Les résultats tests utilisateurs effectués par les deux élèves de masters ont été plutôt positifs.

Nous avons un peu négligé le rendu, préférant nous concentrer sur des interactions fonctionnelles. Nous aurons certainement gagné beaucoup dans le cadre du concours en intégrant un autre membre à l'équipe, à l'aise en design et modélisation. Concernant les interactions, nous sommes arrivés au bout de nos objectifs et le projet a été envoyé au 3DUI contest. Nous avons bon espoir d'être au moins sélectionné, car nous faisons confiance au système que nous avons développé.

Il reste cependant quelques améliorations possibles qui permettraient de peaufiner le projet et pourquoi pas de concourir pour Laval 2013 !